FORENSICS 526

Hands On | Six Days | Laptop Required | 36 CPEs

FOR526: Memory Forensics In-Depth

Digital Forensics and Incident Response (DFIR) professionals view the acquisition and analysis of physical memory as critical to the success of an investigation, be it a criminal case, employee policy violation, or enterprise intrusion. Investigators who do not look at volatile memory are *leaving evidence on the table*. The valuable contents of RAM hold evidence of user actions as well as evil processes and furtive behaviors implemented by malicious code. It is this evidence that often proves to be the smoking gun that unravels the story of what happened on a system.

FOR526 provides the critical skills necessary for digital forensics examiners and incident responders to deftly analyze captured memory images and live response audits. By using the most effective freeware and open-source tools in the industry today and delivering a deeper understanding of how these tools work, this five-day course shows DFIR professionals how to unravel the real story of what happened on a system. It is a critical course for any serious investigator who wants to tackle advanced forensics, trusted insider, and incident response cases.

Just as it is crucial to understand disk and registry structures to substantiate findings in traditional system forensics, it is equally critical to understand memory structures. Having in-depth knowledge of Windows memory internals allows the examiner to access target data specific to the needs of the case at hand.

There is an arms race between analysts and attackers. Modern malware and post-exploitation modules increasingly employ self-defense techniques that include more sophisticated rootkit and anti-memory analysis mechanisms that destroy or subvert volatile data. Examiners must have a deeper understanding of memory internals in order to discern the intentions of attackers or rogue trusted insiders.This course draws on best practices and recommendations from experts in the field to guide DFIR professionals through acquisition, validation, and memory analysis with hands-on, real-world, and malware-laden memory images.

FOR526:Memory Forensics In-Depth will teach you:

- **Proper Memory Acquisition:** Demonstrate targeted memory capture ensuring data integrity and combating anti-acquisition techniques
- **How to Find Evil in Memory:** Detect rogue, hidden, and injected processes, kernel-level rootkits, Dynamic Link Libraries (DLL) hijacking, process hollowing, and sophisticated persistence mechanisms
- **Effective Step-by-Step Memory Analysis Techniques:** Use process timelining, high-low level analysis, and walking the Virtual Address Descriptors (VAD) tree to spot anomalous behavior
- **Best Practice Techniques:** Learn when to implement triage, live system analysis, and alternative acquisition techniques and how to devise custom parsing scripts for targeted memory analysis

Remember: "Malware can hide, but it must run." It is this malware paradox that is the key to understanding that while intruders are becoming more advanced with anti-forensic tactics and techniques, it is impossible for them to hide their footprints completely from a skilled incident responder performing memory analysis. FOR526 will ensure that you and your team are ready to respond to the challenges inherent in DFIR by using cutting-edge memory forensics tools and techniques.

Who Should Attend

- Incident response team members
- Law enforcement officers
- Forensic examiners
- Malware analysts
- Information technology professionals
- System administrators
- Anyone who plays a part in the acquisition, preservation, forensics, or analysis of Microsoft Windows computers

You Will Be Able To

- Utilize stream-based data parsing tools to extract AES-encryption keys from a physical memory image to aid in the decryption of encryption files & volumes such as TrueCrypt & BitLocker
- Gain insight into the current network activity of the host system by retrieving network packets from a physical memory image and examining them with a network packet analyzer
- Inspect a Windows crash dump to discern processes, process objects and current system state at the time of the crash through the use of various debugging tools such as kd, WinDBG, and livekd
- Conduct Live System Memory Analysis with the powerful SysInternal's tool, Process Explorer, to collect real-time data on running processes allowing for rapid triage
- Use the SIFT workstation and in-depth knowledge of PE File modules in physical memory, extract and analyze packed and non-packed PE binaries from memory, and compare them to their known disk-bound files
- Discover key features from memory such as the BIOS keyboard buffer, Kernel Debugging Data Block (KDBG), Executive Process (EPROCESS) structures, and handles based on signature and offset searching, gaining a deeper understanding of the inner workings of popular memory analysis tools
- Analyze memory structures using high-level and low-level techniques to reveal hidden and terminated processes and extract processes, drivers, and memory sections for further analysis
- Use a variety of means to capture memory images in the field, explaining the advantages and limitations of each method

digital-forensics.sans.org

SANS *To register, visit sans.org*
or call 301-654-SANS (7267)

For schedules, course updates, prerequisites, special notes, or laptop requirements, visit sans.org/courses

Course Day Descriptions

526.1 HANDS ON: Foundations in Memory Analysis and Acquisition

Simply put, memory analysis has become a ***required skill*** for all incident responders and digital forensics examiners. Regardless of the type of investigation, system memory and its contents often expose the first piece of the evidential thread that, when pulled, unravels the whole picture of what happened on the target system. Where is the malware? How did the machine get infected? Where did the attacker move laterally? Or what did the disgruntled employee do on the system? What lies in physical memory can provide answers to all of these questions and more.

Topics: Why Memory Forensics?; Investigative Methodologies; The Ubuntu SIFT Workstation; The Volatility Framework; System Architectures; Triage versus Full Memory Acquisition; Physical Memory Acquisition

526.2 HANDS ON: Unstructured Analysis and Process Exploration

Structured memory analysis using tools that identify and interpret operating system structures is certainly powerful. However, many remnants of previously allocated memory remain available for analysis, and they cannot be parsed through structure identification. What tools are best for processing fragmented data? Unstructured analysis tools! They neither know nor care about operating system structures. Instead, they examine data, extracting findings using pattern matching. You will learn how to use Bulk Extractor to parse memory images and extract investigative leads such as email addresses, network packets, and more.

Topics: Unstructured Memory Analysis; Page File Analysis; Exploring Process Structures; List Walking and Scanning; Pool Memory; Exploring Process Relationships; Exploring DLLs; Kernel Objects

526.3 HANDS ON: Investigating the User via Memory Artifacts

An incident responder (IR) is often asked to triage a system because of a network intrusion detection system alert. The Security Operations Center makes the call and requires more information due to outbound network traffic from an endpoint and the IR team is asked to respond. In this section, we cover how to enumerate active and terminated TCP connections – selecting the right plugin for the job based on the OS version.

Topics: Network Connections; Virtual Address Descriptors; Detecting Injected Code; Analyzing the Registry via Memory Analysis; User Artifacts in Memory

FOR526 Training Formats

(subject to change)

Live Training
sans.org/security-training/by-location/all

OnSite
sans.org/onsite

vLive Events
sans.org/vlive

Simulcast
sans.org/simulcast

OnDemand
sans.org/ondemand

SelStudy
sans.org/selfstudy

526.4 HANDS ON: Internal Memory Structures (PART I)

Day 4 focuses on introducing some internal memory structures (such as drivers), Windows memory table structures, and extraction techniques for portable executables. As we come to the final steps in our investigative methodology, "Spotting Rootkit Behaviors" and "Extracting Suspicious Binaries," it is important to emphasize again the rootkit paradox. The more malicious code attempts to hide itself, the more abnormal and seemingly suspicious it appears. We will use this concept to evaluate some of the most common structures in Windows memory for hooking, the IDTs and SSDTs.

Topics: Interrupt Descriptor Tables; System Service Descriptor Tables; Drivers; Direct Kernel Object Manipulation; Module Extraction

526.5 HANDS ON: Internal Memory Structures (PART II) and Memory Analysis Challenges

Sometimes an investigator's luck runs out and he or she does not complete a memory acquisition before the target system is taken offline or shut down. In these cases, where else can system memory captures be found? Hibernation files and Windows crashdump files can be valuable sources of information, regardless of whether or not you find yourself with a current memory capture. This section covers the structure of the hibernation and crashdump files, as well as how to convert both into raw memory images that can easily be parsed using Volatility and other tools in our memory forensics weapons arsenal. In addition, we will analyze a crash dump file, discovering just how Windows responds and what information is captured when a system crashes.

Topics: Hibernation Files; Crash Dump Files; Memory Analysis Challenges

526.6 HANDS ON: Final Day Memory Analysis Challenges

This final section provides students with a direct memory forensics challenge that makes use of the SANS NetWars Tournament platform. Your memory analysis skills are put to the test with a variety of hands-on scenarios involving hibernation files, Crash Dump files, and raw memory images, reinforcing techniques covered in the first five sections of the course. These challenges strengthen the students' ability to respond to typical and atypical memory forensics challenges from all types of cases, from investigating the user to isolating the malware. By applying the techniques learned earlier in the course, students consolidate their knowledge and can shore up skill areas where they feel they need additional practice.

Topics: Malware and Rootkit Behavior Detection; Persistence Mechanism Identification; Code Injection Analysis; User Activity Reconstruction; Linux Memory Image Parsing; Mac OSX Memory Image Parsing; Windows Hibernation File Conversion and Analysis; Windows Crash Dump Analysis (Using Windows Debugger)